Two-Sample Hypothesis Testing for Random Graphs

by

Erin E. L. Hunt

A thesis submitted to The Johns Hopkins University
in conformity with the requirements for the degree of
Master of Science in Engineering

Baltimore, Maryland
May, 2019

Abstract

Hypothesis testing for latent position random graphs is a growing area of research, particularly motivated by needs in areas such as neuroscience, fraud detection, and social networks. We explore two problems of statistical inference. Currently, methods such as adjacency spectral embedding (ASE) are used to create test statistics for random graphs. The first chapter of our study presents non-metric multidimensional scaling as an alternative to ASE. We show our procedure is functional for both simulated data and for graphs generated from MRI scans. In the second chapter we explore classical applications of statistical inference in a multi-graph setting. We will isolate important vertices across a set of graphs, and then determine correlations between the important vertices and physical vertex features. We use the same MRI data from Chapter 1. The overall goal of these studies is to test new concepts of statistical inference on graphs via simulations and explorations of real-world data.

Reader: Dr. Minh Tang

Acknowledgments

I would like to thank my thesis advisor, Dr. Minh Tang, for guiding me throughout this process. His enthusiasm and patience have made this a fun and rewarding experience. I want to extend my gratitude to Dr. Youngser Park for providing data and answering all my questions, and to Dr. Carey Priebe for his guidance, suggestions, and Statistical Theory course. Finally, I thank my dad, Katie Hunt, Jamie Hunt, and Ned Cunningham. This last year and a half has been immensely challenging, and I am eternally grateful for your support.

Table of Contents

Table of Contents iv

List of Tables vii

List of Figures ix

1 Introduction 1

1.1 Definitions 5

1.1.1 Latent position random graph 5

1.1.2 Adjacency Spectral Embedding 6

1.1.3 Non-metric Multidimensional Scaling 7

1.1.4 Two test statistics: Frobenius Norm and Procrustes Distance 8

1.1.5 Mouse Connectome Data 9

1.2 Methods 10

1.2.1 Method for Simulated Data 10

1.2.2 Method for Connectome Data 13

1.2.3 Program for Simulated data 14

1.2.4 Program for Mouse Connectome data 15
1.3 Results 16
1.3.1 Simulation 1: X logistic function, Y logistic function, varying ϵ, no scale factor 16
1.3.2 Simulation 2: X logistic function, Y logistic function, constant ϵ, varying Y scale factor 17
1.3.3 Simulation 3: X Gaussian link function, Y Gaussian link function, varying ϵ, no scale factor 18
1.3.4 Simulation 4: X Gaussian link function, Y Gaussian link function, constant ϵ, varying Y scale factor 18
1.3.5 Simulation 5: X Gaussian link function, Y logistic function, varying ϵ, no scale factor 19
1.3.6 Mouse Connectome data 20
1.4 Conclusion 21
1.5 References 21

2 Multi-Graph Case Study **23**
2.1 Introduction 23
2.2 Definitions 24
2.2.1 Omnibus Embedding 24
2.2.2 Classical Multidimensional scaling 26
2.2.3 Multivariate Analysis of Variance (MANOVA) 27
2.2.4 Gaussian Mixture Modeling and Adjusted Rand Index 28

2.2.4.1 Non-parametric tests 30
2.3 Methods and Results 30
2.3.1 Vertex Removal 30
2.3.2 Vertex Features: Method and Results 33
2.3.2.1 Volume 33
2.3.2.2 Hemisphere 34
2.3.2.3 Spatial Coordinates 34
2.4 Conclusion 35
2.5 References 36

List of Tables

1.1 Power of the Frobenius norm test statistic at significance level $\alpha = 0.05$. As ϵ increases, the power of the test increases. As the graph increases in size, the power approaches 1.00 faster. . . . 16

1.2 Power of the Procrustes difference test statistic at significance level $\alpha = 0.05$. We observe the same trends as in Table 1.1. Note, the Procrustes is slower to increase in power. 17

1.3 Power of the Frobenius norm test statistic at significance level $\alpha = 0.05$. As the scale factor increases, but ϵ remains constant, the power of the Frobenius norm approaches 1.00. 17

1.4 Power of the Procrustes difference test statistic at significance level $\alpha = 0.05$. As the scale factor increases, but ϵ remains constant, the power of the Procrustes statistic remains close to 0.05. A scaling factor may impact the power of the test statistic on larger graphs. 18

1.5 Power of the Frobenius norm test statistic at significance level $\alpha = 0.05$. 18

1.6 Power of the Procrustes distance test statistic at significance level $\alpha = 0.05$. 18

1.7 Power of the Frobenius norm test statistic at significance level $\alpha = 0.05$. The ϵ factor remains constant, but the scaling factor of the link function increases. 19

1.8 Power of the Procrustes difference test statistic at significance level $\alpha = 0.05$. The ϵ factor remains constant, but the scaling factor of the link function increases. 19

1.9 Power of the Frobenius norm distance test statistic at significance level $\alpha = 0.05$. 19

1.10 Power of the Procrustes distance test statistic at significance level $\alpha = 0.05$. 20

1.11 The p-values of a permutation test using a Frobenius norm test statistic. If we set the significance level to $\alpha = 0.5$, we reject the null that scan labels are exchangeable. 20

1.12 The p-values of a permutation test using a Procrustes difference test statistic. If we set the significance level to $\alpha = 0.5$, we reject the null that scan labels are exchangeable in some cases only. There may be sufficient similarity between C57 and DB2. . . . 20

List of Figures

2.1 Vertex MANOVA p-values are sorted and displayed on a log-10 scale. The green points are adjusted for multiple comparisons; The red line represents a significant level of 5%. 27

2.2 100 Most Significant Vertices 28

2.3 We apply CMDS to the embedded coordinates and cluster the results. The 24 scans form clusters according to their genotype 32

2.4 After removing the top 100 most significant vertices, roughly, the membership of scans shifts between the three clusters. Removing vertices diminshes our ability to corrrectly classify a set of embedded coordinates, $\hat{X}_i$ to its genotype 33

2.5 There is a large variation of volume size: 157 are less than 1, 165 are between 1 and 10, and 20 are outliers ranging from 10 to 1712. We grouped the vertices by volume size and ran tests on each group. The table shows how the HSIC and MGC p-values differ between groups. In all but one case we fail to reject H_0. . 34

2.6 3-D plot of vertex regions on XYZ plane. Red vertices correspond to the 50 vertices with the most significant MANOVA p-values. 35

Chapter 1

Introduction

The classical problem of two-sample hypothesis testing is relatively new in the setting of latent position random graphs. Previous research outlines theoretical and practical applications of two-sample hypothesis testing (Tang et al., 2017, Ghoshdastier et al., 2017, Ginestet et al., 2017). Tang et al., which presents the first study on two-sample testing, outlines a procedure based on the adjacency spectral embedding (ASE) of two random dot product graphs. Our goal is to expand the scope of this hypothesis test by exploring a framework using non-metric multidimensional scaling in lieu of ASE and by examining other link functions beyond the dot product. Using simulated data and graph networks from a mouse connectome, we demonstrate the feasibility of our methods.

We consider the setting of two unweighted, undirected latent position random graphs with a known vertex correspondence. The latent positions of the two graphs are i.i.d. drawn from two unknown distributions. The goal of our two-sample hypothesis test is to determine if these two latent positions

are equal. Our statistical test is semi-parametric in the sense we assume the latent positions do not originate from a non-parametric distribution. However our test concerns comparing the two latent positions without estimating the parameters of the underlying distributions.

In Tang et al., the authors develop a test statistic based on the spectral embeddings of two graphs' adjacency matrices. With this method, the two graphs are represented in low-dimension Euclidean space and the test statistic is a distance measure of their new sets of coordinates. The consistency, asymptotic normality, and robustness of the ASE method to estimate latent positions has been previously studied (Athreya et al., 2018). Using a bootstrapping procedure, Tang et al. demonstrates the increasing power of a distance-based test statistic as the differences between the underlying distributions increase. Building on this concept of a distance metric based on estimated latent positions, we provide two new contributions.

First, we show non-metric multidimensional scaling (NMDS) is another viable method of reducing a high-dimension adjacency matrix to a set of low-dimension coordinates suitable for statistical inference. In any latent position graph, the matrix of latent position vectors, $\mathbf{X}$, are not observable, but via ASE of the adjacency matrix, the latent positions can be estimated. Along the same line, we propose NMDS to represent the latent positions in d-dimensions and use the resulting coordinates to build a test statistic. The likelihood of an edge in a latent position graph is dependent upon a distance measure; abstractly speaking, a greater distance represents a smaller likelihood of a connection. Because NMDS is a dimension reduction technique based on

preserving ordinality of distances between data points, it may be well-suited to a latent position graph problem. Unlike the ASE method, which is based on an eigendecomposition of the adjacency matrix, NMDS uses an iterative technique based on distance measures. Depending on the initialization, the final set of coordinates may be rotated, flipped, or translated. Consequently, a test-statistic created by NMDS ordination may be able to capture the equality of $\mathbf{X}$ and $\mathbf{Y}$ up to an orthogonal transformation.

Second, we examine new link functions beyond the dot product. In the instance of a random dot product graph (RDPG), the probability of an edge, or link, is determined by a probability matrix, $\mathbf{P} = \mathbf{X}\mathbf{X}^{\mathbf{T}}$. Each entry of matrix $\mathbf{P}$ is a Bernoulli random variable. In reality, the link function, dot product or otherwise, is unknown. We show that the method for two-sample hypothesis testing of RDPGs in Tang et al. can also be applied to graphs with different link functions. We explore a logistic link function and a Gaussian link function, with and without scaling. In the case of either link function, the increasing distance between latent positions decreases the likelihood of an edge between two nodes. We further demonstrate the link functions of two graphs do not need to be the same in order to perform a two-sample hypothesis test. Previous studies have assumed the link functions between two graphs are equivalent. By showing they need not be, we propose our test statistic can be more broadly applied.

We apply our methodology to both simulated data and real-world data from mouse connectomes. We will test the viability of both the new logistic link

function and NMDS on simulated data; we test the NMDS method on connectome data, but not the new link function. To simulate test data, we generate latent positions by randomly sampling data from two normal distributions, $\mathbf{F}$ and $\mathbf{G}$, create a probability matrix using one of the new link functions, and then formulate the adjacency matrices based on these probabilities. From here, we use NMDS to "recover" latent positions. NMDS does not directly estimate the latent positions, but provides a representation in lower dimension thereby enabling subsequent statistical inference. Given the NMDS coordinates, $\hat{\mathbf{X}}$ and $\hat{\mathbf{Y}}$, we then calculate their similarities by taking the Procrustes difference. Because Procrustes allows for the scaling and rotation of matrices, it is well-suited as a test statistic for ordination methods such as NMDS. Using a bootstrap procedure, we will calculate the power of the Procrustes distance test statistic as we vary one latent position distribution, $\mathbf{G}$, by a small factor. We will also create a second test statistic: the Frobenius norm of the differences between adjacency matrices, prior to applying NMDS.

We are able to use mouse connectome data to test the capability of the NMDS method. In this setting link functions are unknown. The mouse connectome dataset provides 8 MRI scans from 4 distinct mouse genotypes, labeled: C57, CAST, DB2, and BTBR. We are able to convert each of the 32 scans into distinct graphs. We create a two-sample test by making pairwise comparisons between each of the 4 mice. Using a permutation test we assess whether the latent positions of the genotypes are similar.

In the following sections we will present our methodology and experimental

results. Overall, we intend to expand upon previous studies of statistical inference on two graphs. We present a novel way of representing latent positions by proposing NMDS instead of ASE. Additionally, we show a distance-based test statistic is viable given different sets of link functions aside from the dot product.

1.1 Definitions

1.1.1 Latent position random graph

A random graph is a network whose edges are determined by a random variable. In our study, a connection is the outcome of a Bernoulli trial: an entry of the adjacency matrix, $a_{ij} = 1$ with probability p_{ij}. An entry of the probability matrix $\mathbf{P}$ is created by a link function between latent positions: $p_{ij} = \kappa(x_i, x_j)$. The link function, κ, could be any probabalistic function such that $\kappa(X, X) \rightarrow [0, 1]$. Previous studies have surveyed the random dot product graph, wherein $\mathbf{P} = \mathbf{X}\mathbf{X}^{\mathbf{T}}$. Our study examines the use of either a logistic link function or Gaussian kernel. Both allow for a scaling factor, a.:

$$\text{Logistic: } p_{ij} = 1/(1 + \exp(a|x_i - x_j|^2))$$

$$\text{Gaussian: } p_{ij} = \exp(-a|x_i - x_j|^2)$$

Each row vector, x_i or x_j, of a latent position matrix corresponds to a node, i or j. As the difference between the latent positions increases, the probability of an edge existing between nodes i and j decreases. Overall, the latent positions are unobserved random variables and the link functions are unknown. Because

we only observe adjacency matrix, $\mathbf{A}$, we must infer the latent positions using methods such as ASE or NMDS to make downstream statistical tests.

1.1.2 Adjacency Spectral Embedding

Previous studies of inference on random graphs have used adjacency spectral embedding to estimate latent positions. In general, an adjacency spectral embedding is based on the eigendecomposition of an adjacency matrix: $\mathbf{A} = \mathbf{U_A S_A U_A^T}$. The adjacency spectral embedding in d-dimensions is defined as $\hat{\mathbf{U}}\hat{\mathbf{S}}^{\mathbf{1/2}}$ where $\hat{\mathbf{U}}$ is a $n \times d$ matrix comprised of the top d eigenvectors, and $\hat{\mathbf{S}}$ is the corresponding matrix of singular values. We note that because we take the square root of $\hat{\mathbf{S}}$, the singular values must be non-negative to avoid imaginary numbers. Prior research has shown the ASE method is a reasonably good approximation of latent positions, as the adjacency matrix, $\mathbf{A}$, and underlying probability matrix, $\mathbf{P}$, are similar enough (Oliveira, 2009, Davis and Kahan, 1970). In Oliveira 2009 the author shows the norm of the difference between an $n \times n$ adjacency matrix and the corresponding probability matrix is bounded: $\|\mathbf{A} - \mathbf{P}\| \sim C(\sqrt{n \log n})$, where C is a constant. Furthermore, an application of the Davis- Kahan Theorm shows the subspace spanned by the the d-largest eigenvectors of $\mathbf{U_A}$ is reasonably close to the subspace spanned by the d-largest eigenvectors of $\mathbf{U_P}$ (Davis and Kahan, 1970). Other authors further refine and prove this result (Lu and Peng, 2013, Cape et al., 2017, and Yu et al., 2015). As a consequence of the "closeness" between $\mathbf{A}$ and $\mathbf{P}$, an adjacency spectral embedding of the observable $\mathbf{A}$ is a reasonably good approximation of $\mathbf{P} = \mathbf{XX^T}$. The ASE of $\mathbf{A}$ yields: $\hat{\mathbf{X}} = \hat{\mathbf{U}}\hat{\mathbf{S}}^{\mathbf{1/2}}$. ASE is a well-studied method for

approximating latent positions; we will leverage this in our multi-graph study in Chapter 2. However, Chapter 1 explores the ability of NMDS represent latent positions.

1.1.3 Non-metric Multidimensional Scaling

Non-metric multidimensional scaling (NMDS) is an ordination method which reduces an n-dimensional matrix to a lower d-dimensional space. To begin, NMDS requires a $n \times n$ pairwise distance matrix, D, in which the distance measure can be any user-defined distance. A software program will create an initial ordering of the n-dimensional data in d-dimensions and then calculate the cost via a Stress function.

$$Stress = \sqrt{\sum_{i,j}(d_{ij} - \hat{d}_{ij}) / \sum_{i,j} d_{ij}^2} \tag{1.1}$$

The Stress function measures disagreement between the rank-order difference in the original n-dimensions to the new rank-order differences in d-dimensions. In the above equation, d_{ij} is the distance created by the ordination of data points p_i and p_j in the d-dimensional space. A regression of the original distance matrix on the ordinated distance matrix is calculated; the predicted distances are $\hat{d}_{ij}$. In an optimal arrangement, the predicted distances will be equal to the ordinated distances, which is to say the rank-order of the original distances is preserved. A software program will iteratively arrange the ordinated distances in d-dimensions until the Stress function reaches a small enough value, or until a user-defined stopping point is reached. We acknowledge a couple disadvantages of NMDS. First, we may not reach a

global optimum, depending on initialization. Second, NMDS may be expensive to compute. A final note: while NMDS preserves the rank-order in the final ordination, the final result may need to be rotated, scaled, or translated for the necessary interpretation.

In our experiments, we will use NMDS to reduce two $n \times n$ adjacency matrices, $\mathbf{A}$ and $\mathbf{B}$, to two-dimensions. We will show the resulting $n \times 2$ coordinates can be used for two-sample hypothesis testing.

1.1.4 Two test statistics: Frobenius Norm and Procrustes Distance

We are proposing a two-sample hypothesis framework which uses the norm of a distance measure. In our simulations we create two types test statistics: a Frobenius norm and a Procrustes distance. First, we compute the Frobenius norm of the difference between two adjacency matrices. The Frobenius norm of the difference is defined as:

$$\|\mathbf{A} - \mathbf{B}\|_F = \sqrt{\sum_{i,j} |(a_{ij} - b_{ij})|^2} \tag{1.2}$$

We will build a distribution of Frobenius norm test statistics via the bootstrap method outlined in the Methods section. In general, a large statistic will suggest the two adjacency matrices are not "close."

The second statistic is a Procrustes distance of the two sets of coordinates resulting from NMDS. A Procrustes alignment solves the following:

$$P = \min_{W} \left\|\hat{\mathbf{X}} - \hat{\mathbf{Y}}\mathbf{W}\right\|_F \tag{1.3}$$

In the above equation, $\hat{\mathbf{X}}$ and $\hat{\mathbf{Y}}$ are the $n \times d$ coordinates resulting from NMDS. Due to the iterative nature of NMDS, the two sets of coordinates for each set may need to be transformed such that their configuration is as "similar" as possible. The Procrustes distance finds the optimal orthogonal transformation, W, which will minimize the distance between $\hat{\mathbf{X}}$ and $\hat{\mathbf{Y}}$. In addition to exploring the Procrustes' suitability as a test under an NMDS framework, we also aim to show that a scaling factor within the link function should not impact the power of the test. We will construct a distribution of Procrustes test statistics with a bootstrap procedure. Ultimately, we want to compare the power of the Frobenius and Procrustes tests as we vary distributions $\mathbf{F}$, $\mathbf{G}$, and the scaling factor, a.

1.1.5 Mouse Connectome Data

A connectome is a mapping of neuronal signals within a brain. The Mouse Connectome dataset is comprised of MRI scans from 4 distinct genotypes, with 8 scans per mouse. The 32 scans are converted to 332×332 graphs in which each vertex of the graph represents a region of the brain and an edge indicates a connection between two regions. In the context of a latent position random graph, we might consider that the latent positions of the 32 scans are i.i.d. drawn from a distributions unique to their genotypes.

All graphs have a known vertex correspondence. Furthermore, physical features of the brain, such as the vertex volume and location, are the same from scan to scan. After some initial analyses of the 32 scans, neuroscientists noticed a physiological difference between BTBR and the other three genotypes:

"BTBR mice exhibit a 100% absence of the corpus callosum and a severely reduced hippocampal commissure" (Wahlsten et al., 2003). We exclude BTBR from analyses in Chapter 2, but include BTBR in our Chapter 1 exploration.

1.2 Methods

1.2.1 Method for Simulated Data

In theory, two sets of latent positions positions, $\mathbf{X}$ and $\mathbf{Y}$, are drawn i.i.d. from two distributions to generate two $n \times n$ graphs, $\mathbf{G_x}$ and $\mathbf{G_y}$. In a real-world setting we only observe the adjacency matrices, $\mathbf{A}$ and $\mathbf{B}$. We propose NMDS can be used to represent the latent positions, $\hat{\mathbf{X}}$ and $\hat{\mathbf{Y}}$, such that they are suitable for statistical inference. The method of our two-sample hypothesis bootstrap procedure is outlined below. In the simulated experiments we must artificially create the latent positions and link function.

Step 1. Construct $\mathbf{X}$ and $\mathbf{Y}$ from two bivariate normal distributions; the latent positions, $\mathbf{Y}$, are a small perturbation of $\mathbf{X}$.

$$\mathbf{X} \sim \mathcal{N}(\mathbf{0},\ \mathbf{I_2})$$

$$\mathbf{Z} \sim \mathcal{N}(\mathbf{0},\ \epsilon\mathbf{I_2})$$

$$\mathbf{Y} = \mathbf{X} + \mathbf{Z}$$

Step 2. From the latent positions, create the edge probability matrices, $\mathbf{P}$ and $\mathbf{Q}$, using a logistic link

function or a Gaussian kernel.

$$p_{ij} = 1/(1 + \exp(a|x_i - x_j|^2)) \qquad \text{Logistic}$$

$$p_{ij} = \exp -(a|x_i - x_j|^2) \qquad \text{Gaussian}$$

The value $|x_i - x_j|$ is the pairwise difference of the row vectors in $\mathbf{X}$. We similarly calculate q_{ij} from $|y_i - y_j|$ to create $\mathbf{Y}$. The logistic function is in a sense 'adjusted' as it is a slightly modified version of the original logistic function: $1/(1 + \exp(-|x_i - x_j|^2))$. We require this adjustment to ensure the likelihood of a connection decreases as the distance between latent positions increases.

Step 3. The adjacency matrices, $\mathbf{A}$ and $\mathbf{B}$, are formulated by Bernoulli trials of the probability matrices:

$$a_{ij} = Bernoulli(p_{ij})$$

$$b_{ij} = Bernoulli(q_{ij})$$

Step 4. We repeat these trials $m = 20$ times, and calculate the mean of the 20 trials to create the mean adjacency matrices, $\bar{\mathbf{A}}$, $\bar{\mathbf{B}}$. The values of $\bar{\mathbf{A}}$, $\bar{\mathbf{B}}$ will approach the actual p_{ij}, q_{ij}, but there is still

some level of noise.

Step 5. Compute the Frobenius norm test statistic:

$$T_F = \|\bar{\mathbf{A}} - \bar{\mathbf{B}}\|_F$$

Step 6. Compute "pass-to-rank" function on $\bar{\mathbf{A}}, \bar{\mathbf{B}}$ such that their entries are ordinal

Step 7. We then calculate $1 - \bar{\mathbf{A}}$ and $1 - \bar{\mathbf{B}}$ to convert the "similarities" to "dissimilarities," as is required for NMDS

Step 8. We use an NMDS function from the smacof library in R to calculate the coordinates of the latent positions in 2 dimensions; the results, $\hat{\mathbf{X}}$ and $\hat{\mathbf{Y}}$, are two $n \times 2$ dimension matrices.

Step 9. Calculate the Procrustes difference between the estimated latent positions:

$$T_P = \min_W \left\|\hat{\mathbf{X}} - \hat{\mathbf{Y}}\mathbf{W}\right\|_F$$

Step 10. Repeat Steps 1 - 9 for 200 trials to create distributions of the two test statistics, T_F and T_P.

In the first experiment, we will run the bootstrapping procedure for $\epsilon \geq 0$. When $\epsilon = 0$ we are computing the test-statistic under the null-hypothesis, $H_0 : \mathbf{X} = \mathbf{Y}$. As we increase the value ϵ, we expect to see both T_F and T_P

increase in power. In the second experiment we examine the case in which a scaling factor is incorporated in the link function.

1.2.2 Method for Connectome Data

In the previous experiments, we test $H_0 : \mathbf{X} = \mathbf{Y}$ for simulated data. In the case of the mouse connectome data, we test whether the latent positions of the 4 genotypes are equivalent: $\mathbf{X_{C57}} = \mathbf{X_{CAST}} = \mathbf{X_{DB2}} = \mathbf{X_{BTBR}}$. For each pair of mice, we run a permutation test, for a total of 6 tests. An example with two mice, C57 and CAST, is outlined below. We recall, each genotype has a corresponding list of 8 adjacency matrices originating from the 8 MRI scans:

$$C57.list = [A_{C57_1}, A_{C57_2}, \dots A_{C57_8}]$$

$$CAST.list = [A_{CAST_1}, A_{CAST_2}, \dots A_{CAST_8}]$$

Step 1. Compute an average of the 8 scans within the genotype:

$$\bar{\mathbf{A}}_{\mathbf{C57}} = \sum_{i=1}^{8} A_{C57_i}$$

$$\bar{\mathbf{A}}_{\mathbf{CAST}} = \sum_{j=1}^{8} A_{CAST_j}$$

Step 2. Apply NMDS to $\bar{\mathbf{A}}_{\mathbf{C57}}$ and $\bar{\mathbf{A}}_{\mathbf{CAST}}$. The resulting coordinates are two 332×2 matrices.

Step 3. Given the results of the NMDS procedure, we compute the Procrustes distance between the

coordinates, $\hat{\mathbf{X}}_{\mathbf{C57}}$ and $\hat{\mathbf{X}}_{\mathbf{CAST}}$

$$T_{null} = \min_{W} \left\| \hat{\mathbf{X}}_{\mathbf{C57}} - \hat{\mathbf{X}}_{\mathbf{CAST}} \mathbf{W} \right\|_F$$

Step 4. Permute the scans within the two lists, C57.list and CAST.list, to create two new lists, each of length 8. For example:

$$A.list.perm = [A_{C57_4}, A_{CAST_7}, A_{C57_7}, A_{CAST_3}, \ldots A_{C57_6}]$$

$$B.list.perm = [A_{CAST_6}, A_{C57_5}, A_{C57_1}, A_{CAST_8}, \ldots A_{CAST_4}]$$

Step 5. Compute $\bar{\mathbf{A}}$ and $\bar{\mathbf{B}}$ from the two permuted lists

Step 6. Re-apply NMDS to $\bar{\mathbf{A}}$ and $\bar{\mathbf{B}}$. Re-compute the Procrustes distance on the estimated coordinates

Step 7. Repeat Steps 4 - 6 for 100 (or more) trials to create a distribution of statistics based on the re-arranged lists.

The null hypothesis assumes the scans of the two genotypes are exchangeable. If the test statistic computed in Step 3 falls within the 95th percentile of the distribution created in Step 7, we reject the null.

1.2.3 Program for Simulated data

Pseudo-code for bootstrap function

```
Bootstrap(eps, scale, bs, n):
    While count < bs:
        X <- mvnorm(row = n, Mu = c(0,0), Sigma = diag(2))
        Z <- mvnorm(row = n, Mu = c(0,0), Sigma = eps*diag(2))
        Y <- X + Z
        P <- edge.prob.generator(X, scale)
        Q <- edge.prob.generator(Y, scale)
            for (i in 1:m){
             A_list[[i]] <- rg.sample(P); B_list[[i]] <- rg.sample(Q) }
        A_bar <- mean(A_list); B_bar <- mean(B_list)
        T_f <- norm(A_bar - B_bar, type = "F")
        Abar.mds <- PTR(1-A_bar) ; Bbar.mds <- PTR(1-B_bar)
        Xhat <- NMDS(Abar.mds, type = "ordinal", ndim = 2)
        Yhat <- NMDS(Bbar.mds, type = "ordinal", ndim = 2)
       T_x <- procrustes(Xhat, Yhat); T_y <- procrustes(Yhat, Xhat)
        T_p <- min(T_x, T_y)
```

1.2.4 Program for Mouse Connectome data

Pseudo-code to create a single test statistic given two lists, MouseA and MouseB, of either original or permuted scan labels.

```
connectome_stat(MouseA, MouseB, type):
    A_bar <- mean(MouseA_list); B_bar <- mean(MouseB_list)
    T_f <- norm(A_bar - B_bar, type = "F")
    Abar.mds <- PTR(1-A_bar) ; Bbar.mds <- PTR(1-B_bar)
    Xhat <- NMDS(Abar.mds, type = "ordinal", ndim = 2)
    Yhat <- NMDS(Bbar.mds, type = "ordinal", ndim = 2)
    T_x <- procrustes(Xhat, Yhat); T_y <- procrustes(Yhat, Xhat)
    T_p <- min(T_x, T_y)
```

1.3 Results

1.3.1 Simulation 1: X logistic function, Y logistic function, varying ϵ, no scale factor

In the first experiment we consider the case in which the probability of an edge is determined by a logistic link function: $1/(1+\exp(a|x_i - x_j|^2))$. With 200 bootstrap replicates, we compute distributions for T_F and T_P and illustrate their relative powers when $\alpha = 0.05$ for $\epsilon \in \{0, 0.02, 0.05, 0.1, 0.2, 1\}$. We observe power increases as ϵ increases and power increases at a faster rate when the size of the graph, $n \in \{20, 50, 100\}$, increases as well. Overall, as ϵ and n grow, the power of the Procrustes distance statistic is slower to approach 1 than the Frobenius norm.

	n	ϵ =0	ϵ =0.02	ϵ =0.05	ϵ =0.1	ϵ =0.2	ϵ =1
1	20	0.05	0.12	0.47	0.81	0.98	1.00
2	50	0.05	0.65	0.97	1.00	1.00	1.00
3	100	0.05	0.75	1.00	1.00	1.00	1.00

Table 1.1: Power of the Frobenius norm test statistic at significance level $\alpha = 0.05$. As ϵ increases, the power of the test increases. As the graph increases in size, the power approaches 1.00 faster.

	n	ϵ =0	ϵ =0.02	ϵ =0.05	ϵ =0.1	ϵ =0.2	ϵ =1
1	20	0.05	0.09	0.07	0.12	0.14	0.65
2	50	0.05	0.41	0.96	0.96	0.98	1.00
3	100	0.05	0.06	0.98	0.99	1.00	1.00

Table 1.2: Power of the Procrustes difference test statistic at significance level $\alpha = 0.05$. We observe the same trends as in Table 1.1. Note, the Procrustes is slower to increase in power.

1.3.2 Simulation 2: X logistic function, Y logistic function, constant ϵ, varying Y scale factor

In the second simulation, we use a scaling factor a to create $\mathbf{Q}$ from $\mathbf{Y}$: $q_{ij} = 1/(1 + \exp(a|y_i - y_j|^2))$. Given a constant ϵ factor, but an increasing scaling factor, $a \in (1, 1.01, 1.02, 1.05, 1.1, 1.2, 2)$, we see no change in power for the T_P statistic. This is consistent with the notion NMDS uses rank-ordering and not the magnitude of distances when positioning n-dimensional data in 2 dimensions. Although the scaling factor may increase the probability of a connection, it does not change the ordering. However, as a increases, we observe the power of the Frobenius norm test statistic increasing.

	n	a =1	a =1.01	a =1.02	a =1.05	a =1.2	a =2
1	30	0.05	0.05	0.06	0.04	0.06	0.76
2	50	0.05	0.04	0.06	0.01	0.04	0.96
3	100	0.05	0.05	0.05	0.04	0.04	1.00

Table 1.3: Power of the Frobenius norm test statistic at significance level $\alpha = 0.05$. As the scale factor increases, but ϵ remains constant, the power of the Frobenius norm approaches 1.00.

	n	a =1	a =1.01	a =1.02	a =1.05	a =1.2	a =2
1	30	0.05	0.06	0.03	0.07	0.05	0.07
2	50	0.05	0.04	0.06	0.08	0.07	0.15
3	100	0.05	0.04	0.07	0.04	0.08	0.12

Table 1.4: Power of the Procrustes difference test statistic at significance level $\alpha = 0.05$. As the scale factor increases, but ϵ remains constant, the power of the Procrustes statistic remains close to 0.05. A scaling factor may impact the power of the test statistic on larger graphs.

1.3.3 Simulation 3: X Gaussian link function, Y Gaussian link function, varying ϵ, no scale factor

Simulation 3 is a repetition of Simulation 1 using a Gaussian kernel link function in lieu of the logistic link function. We observe the same trends as in Simulation 1.

	n	ϵ =0	ϵ =0.02	ϵ =0.05	ϵ =0.1	ϵ =0.2	ϵ =1
1	20	0.05	0.81	0.99	1.00	1.00	1.00
2	50	0.05	0.99	1.00	1.00	1.00	1.00
3	100	0.05	1.00	1.00	1.00	1.00	1.00

Table 1.5: Power of the Frobenius norm test statistic at significance level $\alpha = 0.05$.

	n	ϵ =0	ϵ =0.02	ϵ =0.05	ϵ =0.1	ϵ =0.2	ϵ =1
1	20	0.05	0.05	0.08	0.12	0.40	0.99
2	50	0.05	0.98	0.97	0.99	1.00	1.00
3	100	0.05	0.99	0.99	1.00	1.00	1.00

Table 1.6: Power of the Procrustes distance test statistic at significance level $\alpha = 0.05$.

1.3.4 Simulation 4: X Gaussian link function, Y Gaussian link function, constant ϵ, varying Y scale factor

Similar to Simulation 2, when we hold ϵ constant but increase the scaling factor in $\mathbf{Q}$, the power of the Procrustes test statistic remains constant while

that of the Frobenius norm increases.

	n	a =1	a =1.01	a =1.02	a =1.05	a =1.2	a =2
1	30	0.05	0.04	0.03	0.04	0.15	1.00
2	50	0.05	0.04	0.04	0.06	0.23	1.00
3	100	0.05	0.01	0.04	0.03	0.27	1.00

Table 1.7: Power of the Frobenius norm test statistic at significance level $\alpha = 0.05$. The ϵ factor remains constant, but the scaling factor of the link function increases.

	n	a =1	a =1.01	a =1.02	a =1.05	a =1.2	a =2
1	30	0.05	0.04	0.04	0.06	0.04	0.07
2	50	0.05	0.04	0.04	0.05	0.03	0.04
3	100	0.05	0.10	0.05	0.06	0.10	0.20

Table 1.8: Power of the Procrustes difference test statistic at significance level $\alpha = 0.05$. The ϵ factor remains constant, but the scaling factor of the link function increases.

1.3.5 Simulation 5: X Gaussian link function, Y logistic function, varying ϵ, no scale factor

The final simulation uses a Gaussian kernel link function to generate **P** and a logistic link function to generate **Q**. As ϵ increases, the results we observe are consistent with those in simulations 1 and 3.

	n	ϵ =0	ϵ =0.02	ϵ =0.05	ϵ =0.1	ϵ =0.2	ϵ =1
1	20	0.05	0.18	0.33	0.60	0.87	1.00
2	50	0.05	0.32	0.70	0.97	1.00	1.00
3	100	0.05	0.43	0.97	1.00	1.00	1.00

Table 1.9: Power of the Frobenius norm distance test statistic at significance level $\alpha = 0.05$.

	n	ϵ =0	ϵ =0.02	ϵ =0.05	ϵ =0.1	ϵ =0.2	ϵ =1
1	20	0.05	0.12	0.09	0.12	0.14	0.79
2	50	0.05	0.05	0.05	0.07	0.14	0.68
3	100	0.05	0.29	1.00	0.99	0.98	1.00

Table 1.10: Power of the Procrustes distance test statistic at significance level $\alpha = 0.05$.

1.3.6 Mouse Connectome data

We conduct a pairwise comparison of each of the 4 mice genotypes. By permuting labels of scans between genotypes we create two distributions of T_F and T_P.

	C57	CAST	DB2	BTBR
C57	—	0.00	0.00	0.00
CAST		—	0.00	0.00
DB2			—	0.00
BTBR				—

Table 1.11: The p-values of a permutation test using a Frobenius norm test statistic. If we set the significance level to $\alpha = 0.5$, we reject the null that scan labels are exchangeable.

	C57	CAST	DB2	BTBR
C57	—	0.00	0.16	0.01
CAST		—	0.05	0.02
DB2			—	0.08
BTBR				—

Table 1.12: The p-values of a permutation test using a Procrustes difference test statistic. If we set the significance level to $\alpha = 0.5$, we reject the null that scan labels are exchangeable in some cases only. There may be sufficient similarity between C57 and DB2.

1.4 Conclusion

Current methods of two-sample hypothesis testing for latent position random graphs depend on a mapping of adjacency matrices in lower-dimension Euclidean space. We expand current research by presenting NMDS as a new method to represent latent positions. In a simulated-data setting we examine new link functions to generate a probability matrix and we explore the powers of Procrustes distance and Frobenius norm test statistics. Overall, we observe similar trends in power, using both the logistic or Gaussian link functions, or a combination of the two. We find the Procrustes test statistic is robust to a scaling factor in the link function when $\mathbf{X}$ and $\mathbf{Y}$ are i.i.d. drawn from the same distribution. The Frobenius norm test statistic, which is used to compare an average of adjacency matrices, is unable to account for the scaling factor; the power of the test increases despite $\mathbf{X}$ and $\mathbf{Y}$ originating from the same distribution. The simulations using the mouse connectome data suggest that NMDS is a viable method using real-world data. By using NMDS to represent the 4 genotypes' matrices in 2 dimensions, we are able the detect distinctions between genotypes using a permutation test. Explorations of statistical inference on graphs are often motivated by practical needs. In addition to furthering the two-sample hypothesis test of Tang et al., we demonstrate the potential use of NMDS in the field of brain imaging.

1.5 References

D. Ghoshdastidar, M. Gutzeit, A. Carpentier and U. von Luxburg, (2017). Two-Sample Tests for Large Random Graphs using Network Statistics. Arxiv Pre-print: https://arxiv.org/abs/1705.06168

C. E. Ginestet, J. Li, P. Balachandran, S. Rosenberg, and E. D. Kolaczyk, (2017). Hypothesis testing for network data in functional neuroimaging. The Annals of Applied Statistics 11 725-750.

M. Tang, A. Athreya, D. L. Sussman, V. Lyzinski, Y. Park, and C. E. Priebe, (2016). A semiparametric two-sample hypothesis testing problem for random dot product graphs. Journal of Computational and Graphical Statistics.

A. Athreya, D. E. Fishkind, K. Levin, V. Lyzinski, Y. Qin, Y. Park, D. L. Sussman, M. Tang, J. T. Vogelstein, and C. E. Priebe, (2018). Statistical Inference on Random Dot Product Graphs: a Survey. Journal of Machine Learning Research.

Y. Yu, T. Wang, and R. J. Samworth, (2015). A useful variant of the Davis-Kahan theorem for statisticians. Biometrika, 102:315-323.

R. I. Oliveira, (2009). Concentration of the adjacency matrix and of the Laplacian in random graphs with independent edges. Arxiv Pre-print: http://arxiv.org/abs/0911.0600

L. Lu and X. Peng, (2013). Spectra of edge-independent random graphs. Electronic Journal of Combinatorics.

D. Wahlsten, P. Metten, J.C. Crabbe, (2003). Survey of 21 inbred mouse strains in two laboratories reveals that BTBR T/+ tf/tf has severely reduced hippocampal commissure and absent corpus callosum. Brain Res.

Chapter 2

Multi-Graph Case Study

2.1 Introduction

In Chapter 1 we presented new methods for generalizing two-sample hypothesis testing. In Chapter 2 we will present a case-study for multiple-graph hypothesis testing using the Mouse Connectome data. There exist needs for statistical inference on graphs in a biological context. For example, a statistical test to differentiate a healthy patient's MRI scan from that of an Alzheimer's patient. Furthermore, neurologists may be interested in which physical features of the brain lend to these differences. In this case-study we examine which vertices are most important for differentiating one mouse genotype from another.

This case-study examines 3 of the 4 genotypes outlined in chapter 1: C57, CAST, and DB2. For each genotype, we are provided 8 MRI scans from which we generate a total of 24 weighted-edge graphs. If we consider each of the 24 graphs as latent position random graphs, in a sense, each grouping of 8 graphs may be drawn from a unique distribution relating to the genotype. In Chapter

1 we use NMDS and permutation tests to support this claim. In Chapter 2 we return to ASE as a means of estimating latent positions. By embedding the 24 graphs, applying CMDS, and then clustering, we observe 3 distinct groupings of scans for each genotype. Thus, via either the NMDS or ASE method, there exists a distinction based on genotype. Previous studies have used Omnibus embedding for multi-graph hypothesis testing and a downstream MANOVA test to identify vertices that are statistically significant across the graphs (Levin et al., 2019). We leverage these procedures to rank most-significant vertices. Given these rankings, we perform two experiments. First, we observe how physical features of the nodes may correlate with the MANOVA p-values. For each vertex, we are given the volume of the region, the spatial coordinates in three dimensions, and an indicator as to whether the region is in the right or left hemisphere of the brain. Second, we remove significant vertices from each of the 24 graphs to see at what point we are unable to distinguish genotypes from one another.

2.2 Definitions

2.2.1 Omnibus Embedding

We use an omnibus embedding, ASE(O), as an alternative to embedding the average of the 8 graphs within a genotype: ASE($\bar{\mathbf{A}}_{\mathbf{C57}}$), $ASE(\bar{\mathbf{A}}_{\mathbf{CAST}})$, $ASE(\bar{\mathbf{A}}_{\mathbf{DB2}})$. The omnibus method provides a set of coordinates for each vertex of the 24 graphs, enabling us to make statistical inference within and across the genotypes. In the remainder of this section we illustrate the construction and

embedding of an omnibus matrix.

In a simple example, we consider two n-sized vertex-matched graphs and their adjacency matrices, A_1 and A_2. We construct a $2n \times 2n$ omnibus matrix from the adjacency matrices, A_1 and A_2.

$$O = \begin{bmatrix} A_1 & \frac{A_1+A_2}{2} \\ \frac{A_2+A_1}{2} & A_2 \end{bmatrix}$$

We will then compute the ASE of O. Under the null, we assume the latent positions of the two graphs are equivalent.

$$\mathbb{E}(A_1) = \mathbb{E}(A_2) = XX^T = P = U_P S_P U_P^T \tag{2.1}$$

The ASE of O results in the following:

$$\hat{Z} = \begin{bmatrix} \hat{X} \\ \hat{Y} \end{bmatrix}$$

Because the two latent positions are aligned, to create a test statistic we need only calculate the Frobenius norm of their differences, therby avoiding the need to calculate the Procrustes distance: $T_{OMNI} = \left\| \hat{X} - \hat{Y} \right\|_F$.

In the Mouse Connectome experiment, we extend the omnibus example above to a setting of 24 adjacency matrices with a common vertex correspondence: $O \in \mathbb{R}^{(24\times n)\times(24\times n)}$. We compute the ASE of O in d-dimensions, resulting in a $24n \times d$ matrix in which the alignment of the estimated latent positions are preserved, $\hat{Z} = [\hat{X}_1, \hat{X}_2, \ldots, \hat{X}_{24}]^T$. To compute pairwise test statistics between

each of the 24 graphs, we would calculate the following:

$$T_{OMNI} = \left\| \hat{X}_i - \hat{Y}_j \right\|_F \text{ where } i, j \in [1, 24] \tag{2.2}$$

2.2.2 Classical Multidimensional scaling

Classical Multidimensional scaling (CMDS) is a dimension reduction technique in which a matrix of pair-wise distances between data points are represented in a lower dimension. In the reduced dimension space, the orderings of the pair-wise distances are preserved such that similar data will form clusters. To find the optimal orientation in the lower d-dimensional space, the CMDS method finds the eigenspace which represents the greatest variance of the data:

Step 1. Create distance matrix, D, by computing a user-defined pairwise distance measure between data points.

Step 2. Create new matrix, B: the square of each entry of distance matrix, D, followed by a double centering, such that $B = -\frac{1}{2} J D^{(2)} J$ where $J = I - \frac{1}{n}\mathbf{1}\mathbf{1}^{\mathbf{T}}$

Step 3. The solution, X, is given by the d-largest eigenvalues, $\lambda_1 \ldots \lambda_d$ and corresponding eigenvectors, $e_1 \ldots e_d$, of B. The new set of coordinates are as follows: $X = E_d \Lambda_d^{1/2}$.

In the context of our mouse experiment, we create a distance measure by calculating the Frobenius norm of the difference between each of the 24 sub-matrices

from the Omnibus embedding. The resulting matrix, D, is of dimension 24×24, where each entry is calculated as: $d_{ij} = \left\| \hat{X}_i - \hat{Y}_j \right\|_F$ where $i, j \in [1, 24]$. We then apply CMDS to reduce D to $d = 3$ dimensions.

2.2.3 Multivariate Analysis of Variance (MANOVA)

The MANOVA procedure is a statistical test to compare the multivariate sample means of data across multiple categories. Previous studies have used MANOVA to identify vertices whose embedded coordinates are significantly different across groups (Levin et al.). The embedding of the omnibus matrix provides 24 sets of coordinates in $\mathbb{R}^3$ for each vertex. For each of the 332 vertices among the 24 graphs, we apply a MANOVA to assess whether the means of the coordinates are significantly different across the three categories. Using this procedure, we find which of the 332 coordinates may have the most significance in determining the differences between the 3 genotypes.

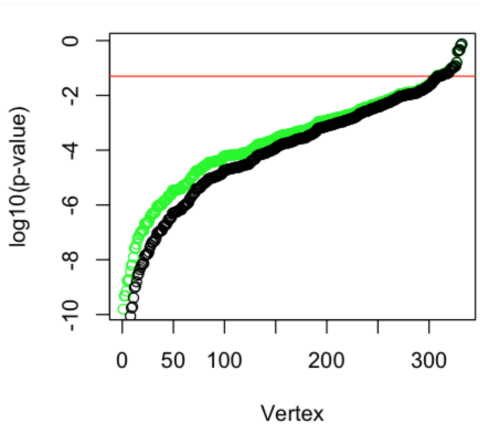

Figure 2.1: Vertex MANOVA p-values are sorted and displayed on a log-10 scale. The green points are adjusted for multiple comparisons; The red line represents a significant level of 5%.

```
 [1] 226 231 248 105 112  44 185 223 314  17 328 144  63 311 130 249  96  51
[19] 143 182  68  16 316 268 298 164  70  60 324 159 325 243 323 194 219 297
[37] 234  65 266 203 273 308 133 184  49 274 229 100 230 228 221 174  43 163
[55] 295 329 318 181  57  91 242 120 114 136 326 215 176 196 123 262  45  13
[73] 288 239 285 236  92  15 235 278 257  41 331 255 214 317 280 180 233 220
[91] 217 313  61 167 256 263 173 227 125 195
```

Figure 2.2: 100 Most Significant Vertices

2.2.4 Gaussian Mixture Modeling and Adjusted Rand Index

We use Gaussian Mixture Modeling (GMM) to cluster the coordinates output by CMDS. The GMM procedure will infer the parameters of k components and the cluster-membership probabilties (the probabilties x_i belongs to cluster k):

$$\theta : (\mu_1, \Sigma_1), ...(\mu_k, \Sigma_k)$$

$$\sum_{i=1}^{k} \kappa_i = 1,$$

In summary, an E-M algorithm begins with an initial set of parameters, θ, then calculates a new set of parameters θ' which will improve the likelihood function, such that $P(X|\theta') > P(X|\theta)$. This will repeat until convergence. For a simple case of 3 clusters, the "E"-step calculates κ for each x_i given current θ. The "M"-step, given k membership probabilities, $\kappa_{k,i}$, will compute the likelihood function. For three clusters, $k = 3$, the probabilities x_i belongs to cluster 1, 2, and 3, are $\kappa_{1,i}$, $\kappa_{2,i}$, and $(1 - \kappa_{1,i} - \kappa_{2,i})$. The likelihood function is written as: $L(\theta; x) = \prod_{i=1}^{n} \left[\kappa_{1,i} G(x_i; \mu_1, \Sigma_1) + \kappa_{2,i} G(x_i; \mu_2, \Sigma_2) + (1 - \kappa_{1,i} - \kappa_{2,i}) G(x_i; \mu_3, \Sigma_3)\right]$.

Using the mclust package in R, we assess the Bayesian Information Criterion (BIC) to select the model's optimal value of k and the best 'Model Name'. The

'Model name' refers to different shapes of clusters, as exemplified below:

EII	spherical, equal volume
VII	spherical, unequal volume
EEI	diagonal, equal volume and shape
VEI	diagonal, varying volume, equal shape
EVI	diagonal, equal volume, varying shape
VVI	diagonal, varying volume and shape
EEE	ellipsoidal, equal volume, shape, and orientation
⋮	⋮

The BIC is defined as: $ln(n)q - 2ln(\hat{L})$, where n = number of data points, q = number of estimated parameters, and $\hat{L}$ refers to the output of the likelihood function after the EM-algorithm has converged.

In this study, we will use the Adjusted Rand Index (ARI) to evaluate the similarity between clusterings as we remove vertices from our graphs. The value of the ARI ranges from 0 to 1, where 1 represents a perfect match of cluster assignments between two models. In general, a Rand Index, RI, is the measure of the frequency of agreements between a set of clusterings over the total number of pairs. The ARI, defined below, adjusts the Rand Index by accounting for alignments between the clusters that may have occured by chance:

$$ARI = \frac{RI - Expected(RI)}{Max(RI) - Expected(RI)}$$

2.2.4.1 Non-parametric tests

To investigate dependence between the vertex MANOVA p-values and the physical vertex features we will use the Hilbert Schmidt Independence Criterion (HSIC) (Gretton et al.) and "Multi-scale Graph Correlation" (MGC) (Shen et al.). Both tests are suited to determining relationships between nonlinear, noisy data. Given a set of pairwise data, $(\mathcal{X}_n, \mathcal{Y}_n) = (x_1, y_1), ..., (x_n, y_n)$, we leverage HSIC and MGC tests to determine if there exists dependence between the two distributions, $\mathcal{X}_n$ and $\mathcal{Y}_n$. In our study, the distribution $\mathcal{X}_n$ refers to vertex MANOVA p-values, while $\mathcal{Y}_n$ is the distribution of the vertex features to be explored; for example, the distribution of the vertices' volume.

2.3 Methods and Results

2.3.1 Vertex Removal

In this study we cumulatively remove the i most significant vertices from the embedded coordinates, where i ranges from 1 to 330. This experiment entails two phases. First we apply ASE to an omnibus matrix to estimate $\hat{\mathbf{X}}$. In the second stage, we cluster the coordinates of $\hat{X}^{(i)}$, where i represents removed vertices. We use an Adjusted Rand Index to compare the results of the second stage to that of the first.

To create the ordered list of vertex MANOVA p-values and then the baseline set of coordinates and clusterings, we compute the following steps

Step 1. For each adjacency matrix, $A_1, ..., A_{24}$, compute $PTR(A_i)$.

Step 2. Given the ordered adjacncy matrices, construct the omnibus representation, $O \in \mathbb{R}^{(24\times 332)\times(24\times 332)}$:

$$O = \begin{bmatrix} A_1 & \dots & (A_1 + A_{24})/2 \\ \vdots & \ddots & \vdots \\ (A_{24} + A_1)/2 & \dots & A_{24} \end{bmatrix}$$

Step 3. Compute $\hat{\mathbf{X}}$ by embedding O in $d = 3$ dimensions. The embedded coordinates, $\hat{\mathbf{X}}$, are a $(332 \times 24) \times (3)$-dimension matrix, with 24 submatrices, each of size 332×3

$$\hat{\mathbf{X}} = \begin{bmatrix} \hat{X}_1 \\ \vdots \\ \hat{X}_{24} \end{bmatrix}$$

Step 4. Calculate MANOVA p-values for each of the 332 vertices.

Step 5. Create distance matrix, D, as $d_{jk} = \left\| \hat{X}_j - \hat{X}_k \right\|_F$ for $j \in (1, 24), k \in (1, 24)$

Step 6. Apply CMDS to D to reduce data to 2 dimensions

Step 7. Cluster the two-dimensional data using GMM to visualize groupings of the embedded coordinates.

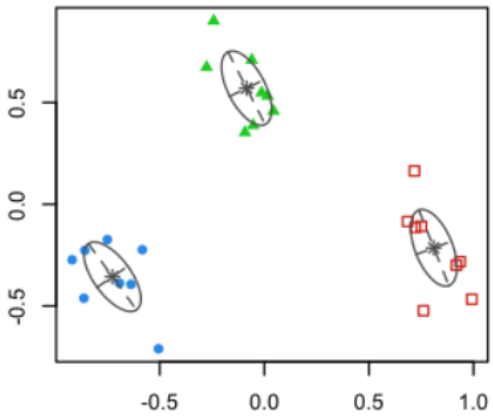

Figure 2.3: We apply CMDS to the embedded coordinates and cluster the results. The 24 scans form clusters according to their genotype

In the second phase, we remove i vertices from $\hat{\mathbf{X}}$, and then re-apply CMDS $\rightarrow$ GMM. We compute an ARI to quantify the changes in cluster assignments as we cumulatively remove the most significant vertices.

Step 1. Remove i from the embedded data, $\hat{\mathbf{X}}$, such that for any $j \in (1, 24)$, $\hat{X}_j^{(i)}$ is the matrix obtained by removing rows of $\hat{X}_j$ corresponding to the i-smallest p-values.

Step 2. Re-compute $D^{(i)}$ as $d_{jk}^{(i)} = \left\| \hat{X}_j^{(i)} - \hat{X}_k^{(i)} \right\|_F$

Step 3. Apply CMDS to D to reduce data to 2 dimensions

Step 4. Cluster the two-dimensional data using GMM to visualize groupings of the embedded coordinates.

Step 5. compute the ARI between the current clustering and the clustering in Phase 1.

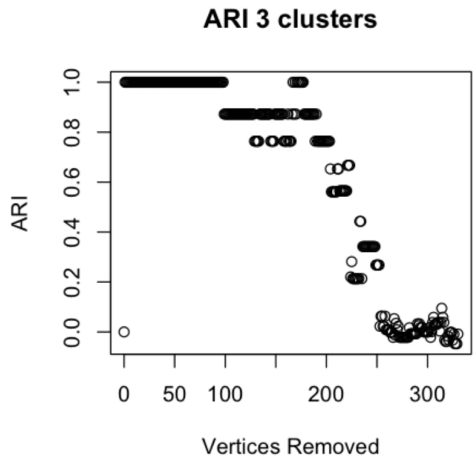

Figure 2.4: After removing the top 100 most significant vertices, roughly, the membership of scans shifts between the three clusters. Removing vertices diminshes our ability to corrrectly classify a set of embedded coordinates, $\hat{X}_i$ to its genotype

2.3.2 Vertex Features: Method and Results

2.3.2.1 Volume

We used both HSIC and MGC to test, respectively, H_0: p-values and volume are jointly independent and H_0: p-values and volume are uncorrelated. Overall, we find that the p-values are not explained by vertex volume. Our results are presented below.

Vol Lower Bound	Vol Upper Bound	Num Vertices	HSIC p-value	MGC p-value
0	1.0	157	0.327	0.287
1	3.0	97	0.050	0.308
3	10.0	58	0.395	0.376
10	20.0	10	0.827	0.687
20	1712.7	10	0.129	0.244

Figure 2.5: There is a large variation of volume size: 157 are less than 1, 165 are between 1 and 10, and 20 are outliers ranging from 10 to 1712. We grouped the vertices by volume size and ran tests on each group. The table shows how the HSIC and MGC p-values differ between groups. In all but one case we fail to reject H_0.

2.3.2.2 Hemisphere

Each vertex has a "right" or "left" hemisphere designation within the brain. We use a K-S test to compare the distribution of p-values between either side of the brain. Given the p-value is less than 0.05, we reject the null that the two were drawn from the same distribution. We also ran an MGC test, which resulted in a p-value of 0.015.

```
        Two-sample Kolmogorov-Smirnov test

data:  left and right
D = 0.1988, p-value = 0.002831
alternative hypothesis: two-sided
```

2.3.2.3 Spatial Coordinates

Both HSIC and MGC tests suggest dependence between the vertices' spatial coordinates and the MANOVA p-values. The p-values of the two tests were 0.006 and 0.002, respectively. In Figure 3, we plot the X,Y,Z coordinates in 3D and highlight the 50 most significant vertices in red. In some instances the red vertices form small clusters, which may be worth exploring.

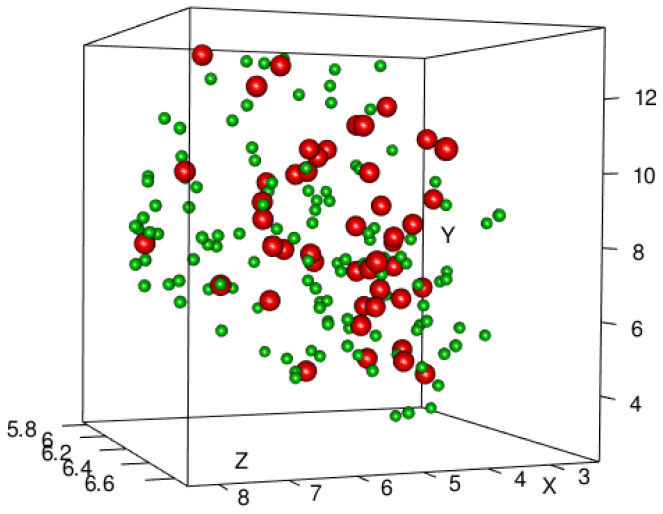

Figure 2.6: 3-D plot of vertex regions on XYZ plane. Red vertices correspond to the 50 vertices with the most significant MANOVA p-values.

2.4 Conclusion

In this case study we demonstrate a practical application of statistical inference on multiple graphs. We identify important vertices across 3 distinct groups of 8 graphs each. We then determined at what point removing the most significant vertices inhibits our ability to differentiate the three groupings. We ran a few simulations, not shown in this paper, in which we cumulatively removed vertices at random; in these preliminary simulations, we needed to remove over 200 vertices before the 3 clusters fell apart. A future project may involve a test that compares structures of clusters given random vertex removal simulations.

Second, given a list of significant vertices and corresponding vertex features,

we make several observations. The MANOVA p-values are not always explained by physical vertex features, as in the case of vertex volume. There may be practical significance in these type of observations. The embedded coordinates may reveal new information that is not apparent from physical observation. Vertex location and MANOVA p-values, however, do appear to exhibit dependence. There may exist opportunities for follow-on studies to explain how graph distance versus physical distance impacts embedding and down-stream statistical inference.

2.5 References

K. Levin, A. Athreya, M. Tang, V. Lyzinski, Y. Park, and C. E. Priebe. (2019) A Central Limit Theorem for an Omnibus Embedding of Random Dot Product Graphs. Arxiv Pre-print: https://arxiv.org/pdf/1705.09355.pdf.

A. Gretton, O. Bousquet, Smola, A. and B. Scholkopf, ÂÍ (2005). Measuring statistical dependence with HilbertâĂŞSchmidt norms. In Algorithmic Learning Theory (S. Jain, H. U. Simon and E. Tomita, eds.). Lecture Notes in Computer Science 3734 63-77.

C. Shen, C. E. Priebe J. T. Vogelstein (2019) From Distance Correlation to Multiscale Graph Correlation, Journal of the American Statistical Association, DOI: 10.1080/01621459.2018.1543125

Erin Elena Lira Hunt

(602) 550-3014 ▪ 3050 Abell Ave ▪ Baltimore, MD 21218 ▪ ehunt8@jhu.edu

Education

Johns Hopkins University, Whiting School of Engineering — **Baltimore, MD**
Master in Applied Mathematics and Statistics — *May 2019*

North Carolina State University, Online & Distance Education — *Jan. - Dec. 2017*
Non-Degree Graduate Coursework in Statistics

Massachusetts Institute of Technology — **Boston, MA**
Bachelor of Science, Economics — *2009 - 2013*

Research

Applied Mathematics & Statistics Department - Johns Hopkins — **Baltimore, MD**
Master's Research for Professor Minh Tang — *Summer 2018 - Present*

- **Simulations.** Developing a test statistic to determine the similarity of two latent-position random graphs generated from different distributions.
- **Practical Applications.** Using mouse MRI scans, whose data form graph networks, I determine the most significant vertices and vertex features. Techniques include multi-dimensional scaling, GMM clustering, MANOVA analysis, and non-parametric statistical tests.

Experience

Freddie Mac (Federal Home Loan Mortgage Corporation) — **McLean, VA**
Data Scientist, Single-Family Risk Analytics — *2015 - 2018*
Second largest purchaser and securitized of U.S. mortgages, $1.9T in home-loan assets (2017)

- **Fraud Modeling.** Built tools to detect fives forms of fraud from FHLMC's 25 million loan database and public records datasets. Directed end-to-end process: discussion of needs with fraud team, research of industry standards, data mining, development of code, and deployment of models to end-users.
- **Collateral Modeling**. Applied random forests to predict physical conditions of homes. Identified data sources in appraisal forms and U.S. tax data. Used n-grams, naive bayes classifiers, and model-selection pipeline to clean home-materials data from free-form text fields.
- **Graph Analytics.** Built graph network to detect anomalous relationships and fraudulent activity between individuals and entities. Employed igraph [R], networkx [Python], and Neo4J database.
- **Distinction.** Received Above-and-Beyond award (company-wide recognition) for creation of Appraiser Monitoring Tool; earned 'Exceeds Expectations' in every performance review (highest designation).
- **Leadership.** Founded monthly book-club; served on committee to organize twice-annual division off-campus activity; prepared PowerPoint deck for and ran monthly team meetings.

CaixaBank, S.A. — **Barcelona, Spain**
Intern, Center for Strategic Development and Innovation — *2013 - 2014*
Spain's largest consumer bank by customer base; €19 billion market capitalization (BME: CABK)

- **Industry Knowledge.** Researched FinTech startup industry developments and maintained deck to inform team of trends in the United States and Europe.
- **Business Development.** Cold-called tech and cryptocurrency startups to broker info-sharing meetings.

Skills

Programming Languages: Python (3+ years), SAS (3+ years), SQL (3+ years), R (2+ year)
Languages: Spanish (proficient)